**TELKOMNIKA**, Vol.16, No.4, August 2018, pp. 1857~1863
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
**DOI:**10.12928/TELKOMNIKA.v16i4.8608


# Development of Leaf Area Meter Using Open CV for Smartphone Application

**Tony K Hariadi*[1], Zulfan Fadholi[2], Anna NN Chamim[3], Nafi A Utama[4], Indira Prabasari[5], Slamet Riyadi[6]**
[1,2,3]Department of Electrical Engineering, Universitas Muhammadiyah Yogyakarta, Jl. Lingkar Barat Tamantirta, Kasihan Yogyakarta, telp +62 274 387656, Indonesia
[4,5]Department of Agrotechnology, Universitas Muhammadiyah Yogyakarta, Jl. Lingkar Barat Tamantirta, Kasihan Yogyakarta, telp +62 274 387656, Indonesia
[6]Department of Information Technology, Universitas Muhammadiyah Yogyakarta, Jl. Lingkar Barat Tamantirta, Kasihan Yogyakarta, telp +62 274 387656, Indonesia
*Corresponding author, e-mail: tonykhariadi@umy.ac.id

***Abstract***

*This study aimed to design an accurate and practical system of leaf area determination using a smartphone. A software application for leaf area computation was developed using Open CV (Open Source Computer Vision) library. Open CV software was tested to estimate the accuracy of leaf area calculation. Leaf area calculations were undertaken using three different image resolutions to compare their accuracy. The results of the software calculations were then compared with the results of the laboratory leaf area meter to identify any errors. The results showed that higher image resolutions improved accuracy by reducing errors. High resolution image gave higher accuracy, however processing speed decreased. Leaf measurement in this project resulted in accuracy range between* 92.8% to 99.0%. *It was concluded that the Open CV algorithm gave fast and adequate accuracy for leaf area calculation, and that the smartphone mobile application system was practical for field use.*

***Keywords***: *Leaf area meter, Open CV, Smartphone application, Leaf area measurement*



## 1. Introduction

Leaf size and its total area can be used to measure plant growth [1],[2]. Leaf area indicates the amount of light that can be absorbed, affecting in the plant productivity and photosynthesis [3]. The concept of Leaf Area Index (LAI) was first introduced in 1947 to demonstrate the relationship between light interception and plant growth [4],[5]. LAI is the ratio between the one-sided leaf area and the ground area under the plant canopy; this shows how much light can be intercepted by the plant for photosynthesis [6]. Leaf area has been measured in various ways, leaf's weight per specific area also taken into account to calculate the product of leaf [7]. Hence, accurate leaf area measurements provide important information to scientists and other researchers.

Image processing has been a popular method for size calculation, color detection, shape detection and other purposes. This method involves processing and calculating different values in digital image [8]. Further numerical process was needed for different objective as mentioned earlier, and shown in the block diagram in Figure 1. Many methods have been developed to calculate leaf area. These methods have evolved to improve calculation accuracy or to improve practical aspects of leaf area meter use. A calculation method using Matlab 6.0 was been developed and the result was compared with grid square method. When tested on several types of leaves, it indicated errors between 3.56%–8.01%, with a better result gained from more squared leaves [9]. Another leaf calculation method from a JPEG file taken using digital camera. The image file was then processed to reduce noise using CIELAB color transformation. After taking 70 samples, the method was found to be 99% accurate, when compared with the grid calculation method [10].

An online method to compute leaf area was also developed. The research made use of Photoshop for image processing comparison. The user uploads the leaf image, which is then converted into 400x400 pixels, with an area of A4 size paper being used as a reference [11].

*Received January 15, 2018; Revised April 11, 2018; Accepted April 28, 2018*

Another computational method used an algorithm programmed in Java to calculate leaf area. The results indicated very high accuracy, however, the calculation process was time consuming [12].

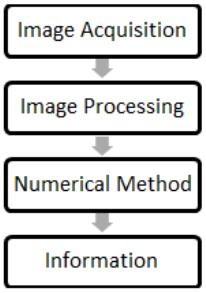



Figure 1. Image Processing Block Diagram

Smartphones are common devices that are equipped with cameras capable of taking good images and embedded processors powerful enough to undertake the calculations associated with leaf area measurement. It is now possible for a mobile application to turn a smartphone into an accessible and practical means of measuring leaf area. The Open CV (Open Source Computer Vision) is a software library for image processing and computer-based vision for any platform. Formerly Open CV was only used with PC's, but now it is also used for the development of Android applications. OpenCV library is free and has been downloaded more than five million times [13].

## 2. Basic Concept and Algorithm

Image processing has many algorithm and preprocessing steps. The steps that has been used in this project were cropping, thresholding, converting and finally calculating the area using Simpson algorithm, Figure 2 indicated the processing step.

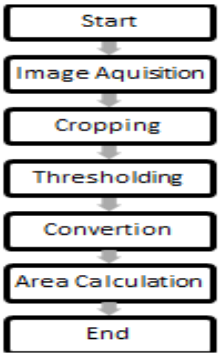



Figure 2. Area Calculation Steps

Capturing image using camera, including smartphone camera was very much dependent to its camera resolution, therefore prior to processing, the image had to be cropped to the desired image resolution and size. In this research design this image area was set to have the size of A4 paper to simplify the calculation. The idea was subtracting the area of A4 with measured leave area, with the basic steps as follows:

1. Set maximum picture area to 210mmx297mm
2. Find total pixel in the area

3. Calculate area value for each pixel in the area

Thresholding was significant step in image processing, to improve image quality when original image was degraded [14]. Many thresholding methods has been developed, including adaptive thresholding. This research project used adaptive thresholding by calculating the mean of a target pixel with its neighborhood. The algorithm was effective when the size of the neighbors was large enough, so that we can assume that the threshold value was the mean of the neighborhood area [15].

T=mean-(maximum-mean) (1)

The next step was to convert the image into gray scale or black and white based on the threshold value. The image was also converted to binary for further calculation. Image conversion was a simple method to isolate the desired image area from the rest of the image. There are many methods in detecting shape and area, including the use of wavelet [16]. Area calculationin this project was done using Simpson method with the following algorithm:

1. Original image was divided into uniform subinterval or windows
2. Set window size as Total Image area/Total Pixel (mm/pixel)
3. Count white pixels
4. Calculate leave area=white pixels x window size

The formula can be simplified as:

$$Area = \frac{a}{r}\, n \quad (2)$$

Where: a is the total image area (210x297 mm),
r : number of pixels in a
n : number of white pixels

## 3. Implementation and Result Pre-performance Test

Calculation performance was also as important as the result itself. Therefore prior to development in Android environment, calculation speed for various image resolutions was tested. Calculation performance for measuring the leaf area was done using personal computer. Different image resolution was performed to find the calculation speed, Figure 3 indicated the result. Various image shapes were also selected to mimic various type of leaves. The result was shown in Figure 2 and indicated that resolutionwas not significant up to 300 dpi which had been a good resolution for many purposes. Above 300 dpi, the calculation speeds were quite difference between various shapes.

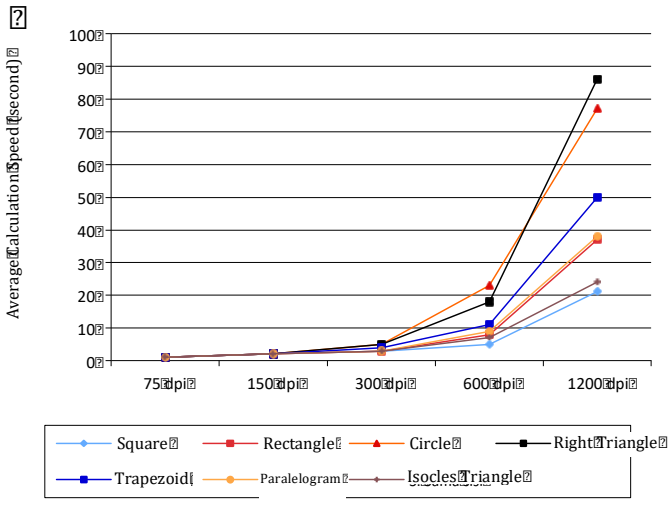



Figure 3. Performance Calculation Test Result

Based on the capability and accuracy provided by Open CV4 Android (docs.opencv.org), a smartphone application was developed for the calculation of leaf areas. This pre-performance calculation also gave good result as follow:

### 3.1. Image Acquisition and Cropping

Firstly, the accuracy of OpenCV calculations in measuring various shapes of a known area was tested. Table 1 shows that the error range of the algorithm was between 2.8%–6.19%, which was sufficient for leaf area calculation. However, small, parallax and measurement errors from using a ruler for the base-line shapes also contributed to the total error calculation.

Using a smartphone camera, images of leaves were taken on A4 size paper to estimate the life size of the leaf. The distance between the smartphone and the paper, was adjusted so that the designated area of the smartphone display was filled completely to the edge of the A4 paper. This gives the best estimation of the actual size of the leaf. Most smartphone display areas are now more than 800x480 pixels or 384,000 pixels. This can be used to represent the size of A4 paper, which is 62.370 mm2. Resolution of 210x297 pixels or 62,370 pixels is needed to use the area of A4 paper as a reference point, since this resolution simply indicated 1mm2 per pixel. When the leaf image is converted to black, the black area is easily counted by the software, where one black pixel equals 1mm$^2$, then the total area of the leaf can be found. Increasing the resolution of the reference area will increase calculation accuracy. For example, the A4 paper can also be represented by 420x594 pixels or 249,480 pixels, where one pixel equals 0.5mm2, and so forth.

Calculating leaf area using Open CV involved sampling and resampling the image to match the resolution of the smartphone display area. This research used three resolutions to compare calculation error. Calculations using the software were compared with readings from a lab's leaf area meter, which was calibrated using a 100x100mm area.

### 3.2. Thresholding

Using adaptive mean thresholding, the code is shown as follow, sample images as the results of thresholding process are shown in Figure 4. Figure 4 demonstrated the result of image captured after thresholding process of A4 paper as the base of the area calculation. Figure 5 demonstrated images of various object on A4 paper.

```
Imgp roc.adaptiveThreshold(gray, purpose, 255, Imgproc.ADAPTIVE_THRESH_MEAN_C,
Imgproc.THRESH_BINARY_INV, 255, 5);
```

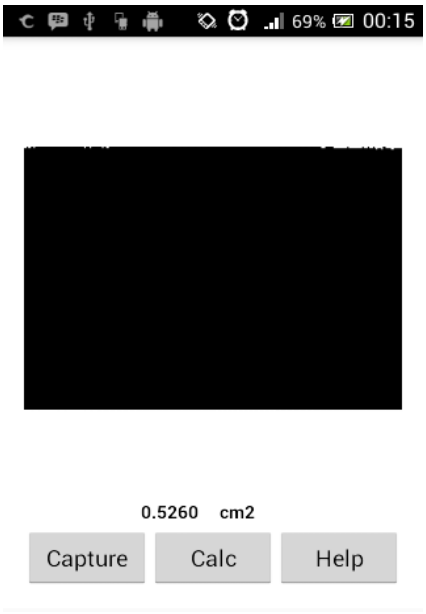



Figure 4. Thresholding result of A4 paper

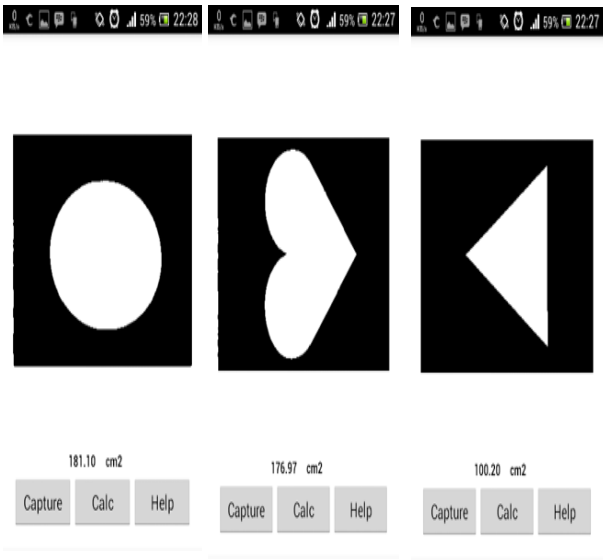



Figure 5. Sample of adaptive thresholding result

### 3.3. Leaf Area Calculation

The calculation can be undertaken in two ways.

1. Indirect method
   a. The leaf image is taken using a smartphone camera then stored as a JPEG image in the phone's storage,
   b. The JPEG image is resampled according to display resolution and
   c. Leaf area calculated.
2. Direct method
   a. A leaf picture is taken using a smartphone camera, and
   b. Leaf area is calculated directly when the image is taken.

During the research, the indirect method, images taken by the smartphone camera were converted into black and white to give better definition between the leaf area and paper areaas shown in Figure 6 and 7. The resolutions that were used for leaf calculations were 210x297 pixels, 420x594 pixels and 820x1188 pixels. These resolutions matched with the aspect ratio of A4 paper and can be accommodated by most smartphone cameras and displays. Calculation code is as follows:

```
Utils.matToBitmap(purpose, bitmap);
iv.setImageBitmap(bitmap);

        Mat m = new Mat();
Core.extractChannel(purpose, m, 0);
intn = Core.countNonZero(m);
intinteger = Integer.parseInt(String.valueOf(n));
// A4 Area in cm2
float a = 623.7f;
//Pixel Value A4 Area in 226x320 resolution
intr = 72320;
double result = (a / r) * integer;
Result.setText(String.valueOf(result));
```

From the code above the value as required in (2) were:
a: 623.7 $cm^2$; r : 72320 pixels; n: calculated leaf area

The same verification method as in the pre-performance test was carried out to evaluate the accuracy of the software. Calculations from the Open CV leaf meter software were compared with the laboratory leaf area meter readings to assess their accuracy, as indicated in Table 3. Higher resolutions gave better calculation accuracy.

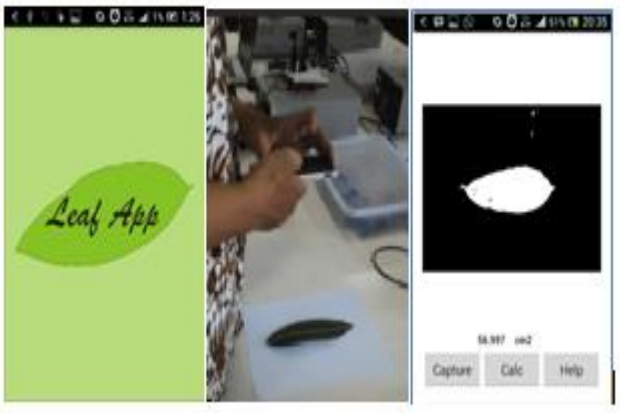



Figure 6 (a), (b), (c). Image aquisition

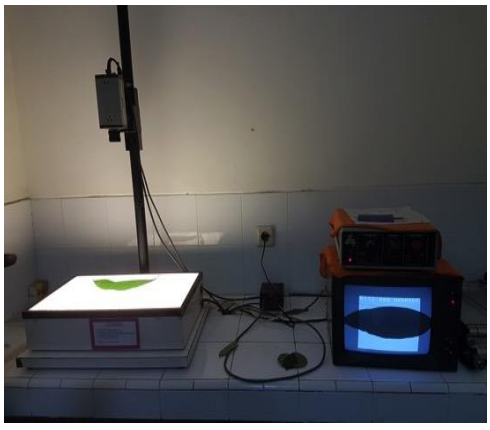


Figure 7. Lab's leaf area meter

Tabel 3. Error Comparisons Between lab's Leaf Area Meter vs Mobile App with Various Image Resolution

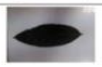
| | Leaf Area meter | OpenCV 210x297 | Error | OpenCV 420x594 | Error | OpenCV 840x188 | Error |
|---|---|---|---|---|---|---|---|
|  | 64 | 54.2 | 15.41 | 56.92 | 11.06 | 58.2 | 9.06 |
|  | 143 | 129.52 | 9.43 | 133.9 | 6.36 | 135.58 | 5.2 |
|  | 22 | 18.84 | 14.36 | 19.74 | 10.27 | 20.37 | 7.41 |
|  | 23 | 20.48 | 10.96 | 21.97 | 4.48 | 22.73 | 1.17 |

Table 4 indicates the measurement result of the application when compared to calculated shape area.

Table 4. Error Comparison Test

| Shape | | Area | | Difference | Error (%) |
|---|---|---|---|---|---|
| | | Calculated (cm2) | *Leaf App* (cm2) | | |
| Square | Sample 1 | 468 | 434,6 | 33,4 | 7,14 |
| | Sample 2 | 234 | 234,7 | 0,7 | 0,3 |
| | Sample 3 | 117 | 119,7 | 2,7 | 2,3 |
| Ellips | Sample 1 | 367,6 | 362,6 | 5 | 1,36 |
| | Sample 2 | 183,8 | 183,2 | 0,5 | 0,27 |
| | Sample 3 | 91,9 | 94,2 | 2,3 | 2,5 |
| Star | Sample 1 | 100,17 | 101,5 | 1,4 | 1,39 |
| | Sample 2 | 56,3 | 56,5 | 0,3 | 0,53 |
| | Sample 3 | 25,0 | 25,4 | 0,4 | 1,6 |
| Round | Sample 1 | 254,5 | 254,6 | 0,1 | 0,04 |
| | Sample 2 | 143,1 | 145,6 | 2,5 | 1,74 |
| | Sample 3 | 63,6 | 64,9 | 1,3 | 2,04 |
| Spade | Sample 1 | 177,3 | 175,4 | 1,9 | 1,07 |
| | Sample 2 | 99,7 | 101.9 | 2,2 | 2,2 |
| | Sample 3 | 44,3 | 45,4 | 1,1 | 2,48 |
| Triangle | Sample 1 | 140,4 | 139,2 | 1,2 | 0,85 |
| | Sample 2 | 105,4 | 103,9 | 1,5 | 1,42 |
| | Sample 3 | 35,1 | 36,5 | 1,4 | 3,99 |

## 4. Conclusion

Software developed for smartphones with the Open CV can be used as a practical means of determining leaf area in the field, while maintaining adequate accuracy. Higher resolutions could improve measurement accuracy, however calculation speed decreased. Smartphone application in this project resulted in high accuracy with between 92.8% to 99.0% while maintaining high speed calculation processing.

## References


[1] Sitompul SM, B Guritno. *Analisis Pertumbuhan Tanaman*, UGM-Press, Yogyakarta. 1995.
[2] Ewert, Frank. Modelling Plant Responses to Elevated CO2: How Important is Leaf Area Index? *Annals of Botany*. 2004; 93: 619-627
[3] Castillo OS, Zaragoza EM, Alvarado CJ, Barrera MG, N Dasgupta-Schubert. Foliar area measurement by a new technique that utilizes the conservative nature of fresh leaf surface density. *International Agrophysics*. 2014; 28(4): 413-421.
[4] Breda, Nathalie JJ. Ground-based measurements of leaf area index: a review of methods, instruments and current controversies. *Journal of Experimental Botany*. 2003; 54(392): 2403-2417.
[5] Watson DJ. Comparative physiological studies in the growth of field crops. I. Variation in net assimilation rate and leaf area between species and varieties, and within and between years. *Annals of Botany*. 1947; 11: 41–76.
[6] Watson DJ. The dependence of net assimilation rate on leaf area index. *Annals of Botany*. 1958; 22: 37–54.
[7] Wolf DD, EW Carson, RH Brown. Leaf Area Index and Specific Leaf Area Determinations. *American Society of Agronomy*. 1972.
[8] Achmad, E Balza, Kartika Firdausy. *Teknik Pengolahan Citra Digital Menggunakan Delphi.* Ardi Publishing, Yogyakarta. 2005,
[9] Li Z, Ji C, Liu J. *Leaf Area Calculating Based on Digital Image*. In: Li D. (eds) Computer And Computing Technologies In Agriculture, Volume II. CCTA 2007. The International Federation for Information Processing, Springer, Boston, MA. 2008; 259.
[10] Chaudhary P, Godara S, Cheeran AN, AK Chaudhari. Fast and Accurate Method for Leaf Area Measurement. *International Journal of Computer Applications*. 2012; 49(9): 22-25.
[11] Can M, Gursoy O, Akcesme B, FB Akcesme. Leaf Area Assessment by Image Analysis. *Southeast Europe Journal of Soft Computing Open journal System*. 2012; 1(2).
[12] Parmar DK, Ghodasara YR, Patel KP, Patel KV, DR Kathiriya. Estimation of Plant Leaf Area using Java Image Processing Techniques. *International Journal on Recent and Innovation Trends in Computing and Communication*. 2015; 3(2): 848 – 851.
[13] Gregori E. *Developing OpenCV Computer Vision Apps for the Android Platform*. 2013.
[14] Yu-Ting et al. Adaptive thresholding algorithm: Efficient computation technique based on intelligent block detection for degraded document images. *Pattern Recognition*. 2010; 43(9): 3177-3184
[15] ER Davies. Computer and Machine Vision: Theory, Algorithms, Practicalities Fourth Edition, Elsevier. 2012.
[16] Riyadi S et al. *Wavelet-based Feature Extraction Technique for Fruit Shape Classification*. Proceding of the 5th International Symposium on Mechatronics and its application (ISMA). 2008.